Ref #	Hits	Search Query	DBs	Default Operator	Plurals	Time Stamp
S1	1	US20040015872A1	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/01/07 10:37
S2	103	("6342581" "6420526" "6433139" "6448230" "6475753" "4991088" "6277959" "6416973" "6410709" "6399351" "6429000" "6383778" "6395889" "6403337" "5546384" "6029181" "5752253" "6233518" "6438543" "6715136" "6291224" "6313376" "6372898" "6388169" "6392017" "6403860" "6416977" "6444419" "6465629" "6476209" "6373971" "5355479" "6321132" "6028899" "6343375" "6026240" "6274312" "6342215" "6361939" "6426072" "6440663" "6440699" "5822592" "6277611" "6329186" "6333182" "6448043" "5870100" "6424962" "6284515").pn.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/01/07 10:37
S3	2	"6681383".pn.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/01/07 11:04
S4	2	"6678882".pn.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/01/07 11:05
S5	2	"6698012".pn.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/01/07 11:06
S6	2	"5555201".pn.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/01/07 11:09
S7	2	"5390325".pn.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/01/07 11:10

S8	2	"6289502".pn.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/01/07 11:29
S9	2	"6678882".pn.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/01/07 11:30
S10	2	"6151701".pn.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/01/07 11:30
S11	2	"6105072".pn.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/01/07 11:31
S12	2	"6275976".pn.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/01/07 11:32
S13	2	"5699310".pn.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/01/07 11:37
S14	731	((recursive adj cycle$1) or loop$1) with validat$5	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/01/07 11:44
S15	2	S14 and (variant with array$1)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/01/07 11:39
S16	0	S14 and (comparing with value$1 with array$1)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/01/07 11:39
S17	303	S14 and (array$1)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/01/07 11:40

S18	1	S14 and (assign$3 with array$1 with (loop$1 or cycle$1))	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/01/07 11:40
S19	62	S14 and (test$3 with array$1)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/01/07 11:40
S20	58	((recursive adj cycle$1) or loop$1) with test$3 with (bound$2 or unbound$2)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/01/07 11:44
S21	13	verif$4 with (loop$1 or cycle$1 or recursion$1) with (bound$2 or unbound$2)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/01/07 12:38
S22	453	717/124.ccls.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/01/07 12:45
S23	293	717/151.ccls.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/01/07 15:02
S24	92	717/152.ccls.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/01/07 15:02
S25	124	717/153.ccls.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/01/07 15:04
S26	177	717/126.ccls.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/01/07 15:05
S27	137	717/160.ccls.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/01/07 15:05

PORTAL
US Patent & Trademark Office

Subscribe (Full Service) Register (Limited Service, Free) Login

Search: ● The ACM Digital Library ○ The Guide

+author:crocker +author:david

THE ACM DIGITAL LIBRARY

Feedback Report a problem Satisfaction survey

Terms used **crocker** **david**

Found **1** of **148,162**

Sort results by: relevance

Display results: expanded form

Save results to a Binder

Search Tips

□ Open results in a new window

Try an Advanced Search

Try this search in The ACM Guide

Results 1 - 1 of 1

Relevance scale □ ▂ ▃ ▨ ▩

1 An internetwork memo distribution capability—MMDF

David H. Crocker, Edward S. Szurkowski, David J. Farber

November 1979 **Proceedings of the sixth symposium on Data communications**

Full text available: pdf(710.41 KB) Additional Information: full citation, abstract, references, citings, index terms

The advent of packet-switched networks has led to increased use of computers for sending text messages (memos) between people. However, attachment to common carrier or equivalent packet networks is expensive and/or restricted by organizational policies. In addition, use of several networks creates the need for relaying memos between them. The work described here is designed to free users from dependence upon any particular communication environment and to provide a memo distribution facilit ...

Results 1 - 1 of 1

Terms of Usage Privacy Policy Code of Ethics Contact Us

Useful downloads: Adobe Acrobat QuickTime Windows Media Player Real Player

PORTAL
US Patent & Trademark Office

Subscribe (Full Service) Register (Limited Service, Free) Login

Search: ● The ACM Digital Library ○ The Guide

+(verifying +or +validating) +recursion +bounds

THE ACM DIGITAL LIBRARY

Feedback Report a problem Satisfaction survey

Published before March 2001
Terms used **verifying** or **validating** **recursion** **bounds**

Found **1,245** of **111,104**

Sort results by: relevance

Display results: expanded form

Save results to a Binder

Search Tips

□ Open results in a new window

Try an Advanced Search
Try this search in The ACM Guide

Results 1 - 20 of 200 Result page: **1** 2 3 4 5 6 7 8 9 10 next

Best 200 shown

Relevance scale □ ▨ ▨ ▨ ■

1 Efficient tests for top-down termination of logical rules
Jeffrey D. Ullman, Allen Van Gelder
April 1988 **Journal of the ACM (JACM)**, Volume 35 Issue 2

Full text available: pdf(2.32 MB) Additional Information: full citation, abstract, references, citings, index terms, review

Considered is the question of whether top-down (Prolog-like) evaluation of a set of logical rules can be guaranteed to terminate. The NAIL! system is designed to process programs consisting of logical rules and to select, for each fragment of the program, the best from among many possible strategies for its evaluation. In the context of such a system, it is essential that termination tests be fast. Thus, the "uniqueness" property of logical rules is introduced. This property is ...

2 Recursion and dynamic data-structures in bounded space: towards embedded ML programming
John Hughes, Lars Pareto
September 1999 **ACM SIGPLAN Notices , Proceedings of the fourth ACM SIGPLAN international conference on Functional programming**, Volume 34 Issue 9

Full text available: pdf(1.48 MB) Additional Information: full citation, abstract, references, citings, index terms

We present a functional language with a type system such that well typed programs run within stated space-bounds. The language is a strict, first-order variant of ML with constructs for explicit storage management. The type system is a variant of Tofte and Talpin's region inference system to which the notion of sized types, of Hughes, Pareto and Sabry, has been added.

3 An automata-theoretic approach to modular model checking
Orna Kupferman, Moshe Y. Vardi
January 2000 **ACM Transactions on Programming Languages and Systems (TOPLAS)**, Volume 22 Issue 1

Full text available: pdf(458.27 KB) Additional Information: full citation, abstract, references, index terms

In modular verification the specification of a module consists of two part. One part describes the guaranteed behavior of the module. The other part describes the assumed behavior of the system in which the module is interacting. This is called the assume-guarantee paradigm. In this paper we consider assume-guarantee specifications in which the guarantee is specified by branching temporal formulas. We distinguish between two approaches. In the first approac ...

Keyw rds: automata, modular verification, temporal logic

Subscribe (Full Service) Register (Limited Service, Free) Login

Search: ● The ACM Digital Library ○ The Guide

+keyword:loop +keyword:bounds

THE ACM DIGITAL LIBRARY

Feedback Report a problem Satisfaction survey

Published before March 2001
Terms used **loop bounds**

Found **2** of **111,104**

Sort results by: relevance

Display results: expanded form

Save results to a Binder

Search Tips

□ Open results in a new window

Try an Advanced Search
Try this search in The ACM Guide

Results 1 - 2 of 2

Relevance scale □ ▤ ▤ ▨ ▩

1 Parallelization of loops with exits on pipelined architectures
P. Tirumalai, M. Lee, M. Schlansker
November 1990 **Proceedings of the 1990 ACM/IEEE conference on Supercomputing**

Full text available: pdf(1.20 MB) Additional Information: full citation, abstract, references

Modulo scheduling theory can be applied successfully to overlap Fortran DO loops on pipelined computers issuing multiple operations per cycle both with and without special loop architectural support [l, 2, 3]. This paper shows that a broader class of loops - *repeat-until, while,* and loops with more than one exit - where the trip count is not known beforehand, can also be overlapped efficiently on multiple issue pipelined machines. Special features that are required in the architecture, as ...

Keywords: Conditional execution, dependence graphs, loop scheduling, modulo scheduling, performance bounds, pipelined architectures, software pipelining, while loops

2 A reexamination of "Optimization of array subscript range checks"
Wei-Ngan Chin, Eak-Khoon Goh
March 1995 **ACM Transactions on Programming Languages and Systems (TOPLAS)**, Volume 17 Issue 2

Full text available: pdf(638.50 KB) Additional Information: full citation, abstract, references, citings, index terms, review

Jonathan Asuru proposed recently an enhanced method for optimizing array subscript range checks. The proposed method is however unsafe and may generate optimized programs whose behavior is different from the original program. Two main flaws in Asuru's method are described, together with suggested remedies and improvements.

Keywords: backward checks propagation, integer programming, loop guard elimination, safe bound checks optimization

Results 1 - 2 of 2

PORTAL

US Patent & Trademark Office

Subscribe (Full Service) Register (Limited Service, Free) Login

Search: ● The ACM Digital Library ○ The Guide

+keyword:verifying +keyword:recursions

Nothing Found

Your search for **+keyword:verifying +keyword:recursions** did not return any results.

You may want to try an Advanced Search for additional options.

Please review the Quick Tips below or for more information see the Search Tips.

Quick Tips

- Enter your search terms in lower case with a space between the terms.

```
sales offices
```

You can also enter a full question or concept in plain language.

```
Where are the sales offices?
```

- Capitalize proper nouns to search for specific people, places, or products.

```
John Colter, Netscape Navigator
```

- Enclose a phrase in double quotes to search for that exact phrase.

```
"museum of natural history" "museum of modern art"
```

- Narrow your searches by using a + if a search term must appear on a page.

```
museum +art
```

- Exclude pages by using a - if a search term must not appear on a page.

```
museum -Paris
```

Combine these techniques to create a specific search query. The better your description of the information you want, the more relevant your results will be.

```
museum +"natural history" dinosaur -Chicago
```

Terms of Usage Privacy Policy Code of Ethics Contact Us

Useful downloads: Adobe Acrobat QuickTime Windows Media Player Real Player

PORTAL
US Patent & Trademark Office

Subscribe (Full Service) Register (Limited Service, Free) Login

Search: ● The ACM Digital Library ○ The Guide

+keyword:recursion +keyword:bound

THE ACM DIGITAL LIBRARY

Feedback Report a problem Satisfaction survey

Published before March 2001
Terms used **recursion** **bound**

Found **1** of **111,104**

Sort results by: relevance

Display results: expanded form

Save results to a Binder
Search Tips
□ Open results in a new window

Try an Advanced Search
Try this search in The ACM Guide

Results 1 - 1 of 1

Relevance scale

1 Algorithm 457: finding all cliques of an undirected graph
Coen Bron, Joep Kerbosch
September 1973 **Communications of the ACM**, Volume 16 Issue 9
Full text available: pdf(804.95 KB) Additional Information: full citation, references, citings

Keywords: backtracking algorithm, branch and bound technique, cliques, clusters, maximal complete subgraphs, recursion

Results 1 - 1 of 1